UNITED STATES PATENT AND TRADEMARK OFFICE

UNITED STATES DEPARTMENT OF COMMERCE
United States Patent and Trademark Office
Address: COMMISSIONER FOR PATENTS
P.O. Box 1450
Alexandria, Virginia 22313-1450
www.uspto.gov

APPLICATION NO.	FILING DATE	FIRST NAMED INVENTOR	ATTORNEY DOCKET NO.	CONFIRMATION NO.
10/567,851	10/06/2006	Jouni Kytomaa	39700-638N01US/NC40070US	7339

64046 7590 03/02/2010

MINTZ, LEVIN, COHN, FERRIS, GLOVSKY AND POPEO, P.C
ONE FINANCIAL CENTER
BOSTON, MA 02111

EXAMINER	
MITCHELL, DANIEL D	

ART UNIT	PAPER NUMBER
2477	

MAIL DATE	DELIVERY MODE
03/02/2010	PAPER

Please find below and/or attached an Office communication concerning this application or proceeding.

The time period for reply, if any, is set in the attached communication.

Office Action Summary	Application No.	Applicant(s)	
	10/567,851	KYTOMAA ET AL.	
	Examiner	Art Unit	
	DANIEL MITCHELL	2477	

-- The MAILING DATE of this communication appears on the cover sheet with the correspondence address --

Period for Reply

A SHORTENED STATUTORY PERIOD FOR REPLY IS SET TO EXPIRE *3* MONTH(S) OR THIRTY (30) DAYS, WHICHEVER IS LONGER, FROM THE MAILING DATE OF THIS COMMUNICATION.

- Extensions of time may be available under the provisions of 37 CFR 1.136(a). In no event, however, may a reply be timely filed after SIX (6) MONTHS from the mailing date of this communication.
- If NO period for reply is specified above, the maximum statutory period will apply and will expire SIX (6) MONTHS from the mailing date of this communication.
- Failure to reply within the set or extended period for reply will, by statute, cause the application to become ABANDONED (35 U.S.C. § 133). Any reply received by the Office later than three months after the mailing date of this communication, even if timely filed, may reduce any earned patent term adjustment. See 37 CFR 1.704(b).

Status

1)☒ Responsive to communication(s) filed on *2/5/2010*.

2a)☐ This action is **FINAL**. 2b)☒ This action is non-final.

3)☐ Since this application is in condition for allowance except for formal matters, prosecution as to the merits is closed in accordance with the practice under *Ex parte Quayle*, 1935 C.D. 11, 453 O.G. 213.

Disposition of Claims

4)☒ Claim(s) *1-17 and 35-46* is/are pending in the application.

4a) Of the above claim(s) *18-34* is/are withdrawn from consideration.

5)☐ Claim(s) _____ is/are allowed.

6)☒ Claim(s) *1-17 and 35-46* is/are rejected.

7)☐ Claim(s) _____ is/are objected to.

8)☐ Claim(s) _____ are subject to restriction and/or election requirement.

Application Papers

9)☐ The specification is objected to by the Examiner.

10)☒ The drawing(s) filed on *10 February 2006* is/are: a)☒ accepted or b)☐ objected to by the Examiner.

Applicant may not request that any objection to the drawing(s) be held in abeyance. See 37 CFR 1.85(a).

Replacement drawing sheet(s) including the correction is required if the drawing(s) is objected to. See 37 CFR 1.121(d).

11)☐ The oath or declaration is objected to by the Examiner. Note the attached Office Action or form PTO-152.

Priority under 35 U.S.C. § 119

12)☒ Acknowledgment is made of a claim for foreign priority under 35 U.S.C. § 119(a)-(d) or (f).

a)☐ All b)☐ Some * c)☐ None of:

1.☐ Certified copies of the priority documents have been received.

2.☐ Certified copies of the priority documents have been received in Application No. _____.

3.☐ Copies of the certified copies of the priority documents have been received in this National Stage application from the International Bureau (PCT Rule 17.2(a)).

* See the attached detailed Office action for a list of the certified copies not received.

Attachment(s)

1) ☒ Notice of References Cited (PTO-892)
2) ☐ Notice of Draftsperson's Patent Drawing Review (PTO-948)
3) ☐ Information Disclosure Statement(s) (PTO/SB/08) Paper No(s)/Mail Date _____.
4) ☐ Interview Summary (PTO-413) Paper No(s)/Mail Date. _____ .
5) ☐ Notice of Informal Patent Application
6) ☐ Other: _____.

DETAILED ACTION

Response to Amendment

1. Applicant's amendment filed on 2/5/2010 has been entered. Claim 2 is amended. Claims 18-34 are canceled. Claims 1-17 and 35-46 are still pending in this application, with claims 1, 18, and 46 being independent.

2. Applicant's request for reconsideration of the finality of the rejection of the last Office action is persuasive and, therefore, the finality of that action is withdrawn.

Response to Arguments

3. Applicant's arguments, see, filed 2/5/2010, with respect to claims 1, 18, and 46 have been fully considered and are persuasive. The 35 USC 103 rejections of claims 1, 18, and 46 have been withdrawn.

Claim Rejections - 35 USC § 103

4. The following is a quotation of 35 U.S.C. 103(a) which forms the basis for all obviousness rejections set forth in this Office action:

5. The following is a quotation of 35 U.S.C. 103(a) which forms the basis for all obviousness rejections set forth in this Office action:

> (a) A patent may not be obtained though the invention is not identically disclosed or described as set forth in section 102 of this title, if the differences between the subject matter sought to be patented and the prior art are such that the subject matter as a whole would have been obvious at the time the invention was made to a person having ordinary skill in the art to which said subject matter pertains. Patentability shall not be negatived by the manner in which the invention was made.

6. Claims 1-3, 6-8, 12-17, 35-38, 42-45 are rejected under 35 U.S.C. 103(a) as being unpatentable over Parruck et al. (US Publication No. 2001/0009552 A1), hereinafter referred as Parruck in view of Yin (US Patent No. 6,219,728 B1) in view of Jung et al. (US Patent No. 5,954,800), hereinafter referred as Jung.

Regarding claim 1, Parruck teaches a method comprising: allocating each received packet to at least one arrival queue **(fig. 5 teaches allocating a received packet to at least one of the arrival queues of elements 402, 404)**;

scheduling, by a scheduler coupled to the at least one arrival queue, packets from the arrival queue to at least one transfer queue **(par. 56-58, fig. 5 teaches a scheduler coupled to the arrival queues of element 402 and transfer queues in element 301)**;

scheduling packets from the processor queues to be processed, wherein the at least one arrival queue **(arrival queue of element 402)**, the at least one transfer queue **(transfer queue of element 301)**, and the plurality of processor queues **(processor queues of element 450)** are separate queues, wherein the scheduler includes a first quantity N of inputs each corresponding to the at least one arrival queue, the scheduler further including a second quantity M of outputs each corresponding to the at least one transfer queue, wherein the second quantity M is less than or equal to the first quantity N **(par. 56-58, fig. 5 further teaches the scheduler includes a first set of inputs and a second set of outputs where the outputs are less in number than the inputs)**.

However Parruck does not expressly disclose placing each packet in the allocated queue if said arrival queue is not full, otherwise dropping said packet; and placing the packet in the allocated processor queue if said queue is not full, otherwise dropping said packet.

Yin teaches a device for receiving packets as the primary reference (col. 1 lines 21-54). Yin teaches in col. 1 lines 21-54 the concept of discarding packets dropping packets when a selected is determined to be full and at capacity.

Therefore, it would have been obvious to one of ordinary skill in the art at the time of the invention to modify the teachings of Parruck to include discarding packets when a buffer is at capacity. One would be motivated as such in order to eliminate overflow of a packet buffer.

However Parruck and Yin do not expressly disclose responsive to transfer of a packet to a transfer queue, generating an interrupt; and responsive to receipt of an interrupt, allocating the packet from said transfer queue to one of a plurality of processor queues.

Jung teaches in **col. 4 line 66 to col. 5 line 15** generating an interrupt to to a CPU after transferring the packets to a transfer queue (memory). Jung further teaches the CPU retrieves the data from the transfer queue and places it in the processor queues in response to the interrupt.

Therefore, it would have been obvious to one of ordinary skill in the art at the time of the invention to modify the teachings of Parruck and Yin to include

generating interrupt signals. One would be motivated as such in order to indicate to the processor that data is present and ready to be processed.

Regarding claim 2, Parruck teaches wherein packets are received at an input to a plurality of devices, wherein the scheduler is further coupled to the at least one transfer queue **(par. 56-58, fig. 5 teaches a device including a plurality of inputs wherein a scheduler is coupled to a transfer queue of element 301)**.

Regarding claim 3, Parruck teaches wherein at least one device has a plurality Of arrival queues **(par. 56-58, fig. 5 teaches a plurality of arrival queues in elements 402 and 404)**.

Regarding claim 6, Parruck teaches wherein at least one device comprises a plurality of transfer queues **(par. 56-58, fig. 5 teaches a device (element 301) including a plurality of transfer queues)**.

Regarding claim 7, Parruck teaches in par. 69 wherein the number of transfer queues is less than the number of arrival queues **(par. 69 teaches the number of queues is a matter of design choice therefore the number of transfer queues is selected to be less than the arrival queues).**

Regarding claim 8, Parruck teaches wherein the scheduling of packets from the arrival queue to the transfer queue is dependent upon one or more of: the traffic profile **(par. 71 teaches packets are scheduled based on the traffic class).**

Regarding claim 12, Parruck teaches wherein the processor queues are associated with different priorities **(par. 92 teaches the queues are all associated with a priority)**.

Regarding claim 13, Parruck teaches wherein the highest priority queue has the lowest drop probability and the lowest latency **(par. 71 teaches different priorities for packets, one of ordinary skill in the art would be able to reasonably discern the highest priority will have the lowest drop probability and the lowest latency).**

Regarding claim 14, Parruck, Yin, and Jung teach a method as the parent claim.

However Parruck and Yin do not expressly disclose responsive to receipt of the interrupt, to remove a packet from a transfer queue, and to classify the packet.

Jung teaches in **col. 4 line 66 to col. 5 line 15** Jung further teaches the CPU retrieves the data from the transfer queue and places it in the processor queues in response to the interrupt.

Regarding claim 15, Parruck teaches wherein the classification is based on a determination of priority **(par. 71 teaches the classification is based on a priority).**

Regarding claim 16, Parruck teaches wherein the packet is allocated to a processor queue in accordance with a classification of the packet **(par. 71 teaches packets are scheduled to the queues based on the packet classification).**

Regarding claim 17, Parruck, Yin and Jung teaches a method as the parent claim.

However Parruck does not expressly disclose wherein the packet is placed in the allocated processor queue if said queue is not full, otherwise the packet is dropped.

Yin teaches a device for receiving packets as the primary reference (col. 1 lines 21-54). Yin teaches in col. 1 lines 21-54 the concept of discarding packets dropping packets when a selected is determined to be full and at capacity.

Therefore, it would have been obvious to one of ordinary skill in the art at the time of the invention to modify the teachings of Parruck to include discarding packets when a buffer is at capacity. One would be motivated as such in order to eliminate overflow of a packet buffer.

Regarding claim 35, Parruck teaches an apparatus comprising: a processor configured to allocate a received packet to at least one arrival queue **(par. 56-58, fig. 5 teaches allocating a received packet to at least one of the arrival queues of elements 402, 404)**; wherein the processor is configured to schedule packets from the arrival queue to at least one transfer queue; wherein the processor is configured to schedule packets from the processor queues to be processed, wherein the at least one arrival queue **(arrival queue of element 402)**, the at least one transfer queue **(transfer queue of element 301)**, and the plurality of processor queues **(processor queues of element 450)** are separate queues , wherein the scheduler includes a first quantity N of inputs each corresponding to the at least one arrival queue, the scheduler further including a second quantity M of outputs each corresponding to the at least one transfer queue, wherein the second quantity M is less than or equal to the first quantity N **(par. 56-58, fig. 5 further teaches the scheduler includes a first set of inputs and a second set of outputs where the outputs are less in number than the inputs)**.

However Parruck does not expressly disclose configured to place each packet in the allocated queue if said queue is not full, otherwise dropping said packet, wherein configured to place the packet in the allocated processor queue if said queue is not full, otherwise dropping said packet.

Yin teaches a device for receiving packets as the primary reference (col. 1 lines 21-54). Yin teaches in col. 1 lines 21-54 the concept of discarding packets dropping packets when a selected is determined to be full and at capacity.

Therefore, it would have been obvious to one of ordinary skill in the art at the time of the invention to modify the teachings of Parruck to include discarding packets when a buffer is at capacity. One would be motivated as such in order to eliminate overflow of a packet buffer.

However Parruck and Yin do not expressly disclose wherein the processor is responsive to transfer of a packet to a transfer queue, configured to generate an interrupt, wherein the processor is responsive to receipt of an interrupt, configured to allocate the packet from said transfer queue to one of a plurality of processor queues.

Jung teaches in **col. 4 line 66 to col. 5 line 15** generating an interrupt to to a CPU after transferring the packets to a transfer queue (memory). Jung further teaches the CPU retrieves the data from the transfer queue and places it in the processor queues in response to the interrupt.

Therefore, it would have been obvious to one of ordinary skill in the art at the time of the invention to modify the teachings of Parruck and Yin to include

generating interrupt signals. One would be motivated as such in order to indicate to the processor that data is present and ready to be processed.

Regarding claim 36, Parruck teaches comprising a plurality of arrival queues **(par. 56-58, fig. 5 teaches a plurality of arrival queues in elements 402, 404).**

Regarding claim 37, Parruck teaches wherein each arrival queue is associated with a traffic class, each packet being allocated to at least one queue by the processor in accordance with the traffic class of each packet **(par. 71 teaches allocating (scheduling) of packets is based on the traffic class of the packet).**

Regarding claim 38, Parruck teaches comprising a plurality of transfer queues **(par. 56-58, fig. 5 teaches a plurality of transfer queues in element 301)**.

Regarding claim 42, Parruck teaches wherein the processor queues are configured to be associated with different priorities **(par. 92 teaches the queues are all associated with a priority).**

Regarding claim 43, Parruck, Yin, and Jung teach an apparatus as the parent claim.

However Parruck and Yin do not expressly disclose responsive to receipt of the interrupt, to remove a packet from a transfer queue, and to classify the packet.

Jung teaches in **col. 4 line 66 to col. 5 line 15** Jung further teaches the CPU retrieves the data from the transfer queue and places it in the processor queues in response to the interrupt in the appropriate location in memory.

Regarding claim 44, Parruck teaches wherein the processor is configured to allocate the packet to a processor queue in accordance with a classification of the packet **(par. 71 teaches packets are scheduled to the queues according to the traffic classification of the packet)**.

Regarding claim 45, Parruck, Yin, and Jung teach an apparatus as the parent claim.

However Parruck does not expressly disclose wherein the packet is placed in the allocated processor queue if said queue is not full, and otherwise the packet is dropped.

Yin teaches a device for receiving packets as the primary reference (col. 1 lines 21-54). Yin teaches in col. 1 lines 21-54 the concept of discarding packets dropping packets when a selected is determined to be full and at capacity.

Therefore, it would have been obvious to one of ordinary skill in the art at the time of the invention to modify the teachings of Parruck to include discarding

packets when a buffer is at capacity. One would be motivated as such in order to eliminate overflow of a packet buffer.

Regarding claim 46, Parruck teaches a computer-readable storage medium encoded with instructions that, when executed on a computer, perform a process, the process comprising: allocating each received packet to at least one arrival queue **(par. 56-58, fig. 5 teaches allocating a received packet to at least one of the arrival queues of elements 402, 404)**; scheduling, by a scheduler coupled to the at least one arrival queue, packets from the arrival queue to at least one transfer queue **(par. 56-58, fig. 5 teaches a scheduler coupled to the arrival queues of element 402 and transfer queues in element 301)**;

scheduling packets from the processor queues for processing, wherein the at least one arrival queue **(arrival queue of element 402)**, the at least one transfer queue **(transfer queue of element 301)**, and the plurality of processor queues **(processor queues of element 450)** are separate queues, wherein the scheduler includes a first quantity N of inputs each corresponding to the at least one arrival queue, the scheduler further including a second quantity M of outputs each corresponding to the at least one transfer queue, wherein the second quantity M is less than or equal to the first quantity N **(fig. 5 further teaches the scheduler includes a first set of inputs and a second set of outputs where the outputs are less in number than the inputs)**.

However Parruck does not expressly disclose placing each packet in the allocated queue if said arrival queue is not full, otherwise dropping said packet; and placing the packet in the allocated processor queue if said queue is not full, otherwise dropping said packet.

Yin teaches a device for receiving packets as the primary reference (col. 1 lines 21-54). Yin teaches in col. 1 lines 21-54 the concept of discarding packets dropping packets when a selected is determined to be full and at capacity.

Therefore, it would have been obvious to one of ordinary skill in the art at the time of the invention to modify the teachings of Parruck to include discarding packets when a buffer is at capacity. One would be motivated as such in order to eliminate overflow of a packet buffer.

However Parruck and Yin do not expressly disclose responsive to transfer of a packet to a transfer queue, generating an interrupt; and responsive to receipt of an interrupt, allocating the packet from said transfer queue to one of a plurality of processor queues.

Jung teaches in **col. 4 line 66 to col. 5 line 15** generating an interrupt to to a CPU after transferring the packets to a transfer queue (memory). Jung further teaches the CPU retrieves the data from the transfer queue and places it in the processor queues in response to the interrupt.

Therefore, it would have been obvious to one of ordinary skill in the art at the time of the invention to modify the teachings of Parruck and Yin to include

generating interrupt signals. One would be motivated as such in order to indicate to the processor that data is present and ready to be processed.

7. Claims 4 and 5 are rejected under 35 U.S.C. 103(a) as being unpatentable over Parruck, Yin, and Jung in view of Iwata et al. (US Publication No. 2004/0114516 A1), hereinafter referred as Iwata.

Regarding claim 4, Parruck, Yin, and Jung teach a method as the parent claim.

However Parruck, Yin, and Jung do not expressly disclose wherein each arrival queue is associated with a traffic class, each packet being allocated to the at least one queue in accordance with the traffic class of each packet.

Iwata teaches a packet scheduler as the primary reference (abstract). Iwata teaches in par. 32 allocating packets based on the traffic class associated with each packet.

Therefore, it would have been obvious to one of ordinary skill in the art at the time of the invention to modify the teachings of Parruck, Yin, and Jung to include allocating packets based on a traffic class. One would be motivated as such in order to preserve the quality of high priority packets **par. 5.**

Regarding claim 5, Parruck, Yin, and Jung teach a method as the parent claim.

However Parruck, Yin, and Jung do not expressly disclose wherein the traffic class is priority information embedded in the each packet.

Iwata teaches a packet scheduler as the primary reference (abstract). Iwata teaches in par. 32 a traffic class is embedded in the header of each packet for classification.

See similar motivation as claim 4.

8. Claims 9, 10, 11, 39, 40, and 41 are rejected under 35 U.S.C. 103(a) as being unpatentable over Parruck, Yin, and Jung in view of Vinnakota.

Regarding claim 9, Parruck, Yin, and Jung teach a method as the parent claim. However Parruck, Yin, and Jung do not expressly disclose wherein the transfer queue comprises a device level transfer queue and a processor level transfer queue, wherein the device level transfer queue receives packets from the arrival queue, and the processor level transfer queue receives packets from the device level transfer queue.

Vinnakota teaches in col. 5 lines 3-14 a transfer queue with a device level transfer queue and a processor level transfer queue. The RAM 122 serves as the device level transfer queues and the queues 140-150 serves as the processor level transfer queues. Vinnakota further teaches in col. 5 lines 3-14 the processor level queues receives packets from the device level queue.

Therefore, it would have been obvious to one of ordinary skill in the art at the time of the invention to modify the teachings of Parruck, Yin, and Jung to

include generating an interrupt signal when a packet is received at a transfer queue. One would be motivated as such in order minimize the processing performed by a packet processor col. 3 lines 11-23.

Regarding claim 10, Parruck, Yin, and Jung teach a method as the parent claim.

However Parruck, Yin, and Jung do not expressly disclose wherein packets are transferred to the processor level transfer queue from the device level transfer queue whenever there is space in the processor level transfer queue.

Vinnakota teaches a packet processor as the primary reference (col. 3 lines 11-23). Vinnakota teaches in col. 5 lines 3-14 that when available packets are transferred from the device level queue to the processor level transfer queue.

See similar motivation as claim 9.

Regarding claim 11, Parruck teaches wherein packets are never dropped from the transfer queue **(par. 64 teaches packets are not dropped from the transfer queues)**.

Regarding claim 39, Parruck, Yin, and Jung teach an apparatus as the parent claim. However Parruck, Yin, and Jung do not expressly disclose wherein the transfer queue comprises a device level transfer queue and a processor level

transfer queue, wherein the device level transfer queue receives packets from the arrival queue, and the processor level transfer queue receives packets from the device level transfer queue.

Vinnakota teaches in col. 5 lines 3-14 a transfer queue with a device level transfer queue and a processor level transfer queue. The RAM 122 serves as the device level transfer queues and the queues 140-150 serves as the processor level transfer queues. Vinnakota further teaches in col. 5 lines 3-14 the processor level queues receives packets from the device level queue.

Therefore, it would have been obvious to one of ordinary skill in the art at the time of the invention to modify the teachings of Parruck, Yin, and Jung to include generating an interrupt signal when a packet is received at a transfer queue. One would be motivated as such in order minimize the processing performed by a packet processor col. 3 lines 11-23.

Regarding claim 40, Parruck, Yin, and Jung teach an apparatus as the parent claim.

However Parruck, Yin, and Jung do not expressly disclose wherein packets are transferred to the processor level transfer queue from the device level transfer queue whenever there is space in the processor level transfer queue.

Vinnakota teaches a packet processor as the primary reference (col. 3 lines 11-23). Vinnakota teaches in col. 5 lines 3-14 that when available packets are transferred from the device level queue to the processor level transfer queue.

See similar motivation as claim 39.

Regarding claim 41, Parruck teaches wherein packets are never dropped from the transfer queue **(par. 64 teaches packets are not dropped from the transfer queues)**.

Conclusion

9. Any response to this action should be **faxed** to (571) 173-8300 or **mailed** to:

Commissioner of Patents
P.O. Box 1450
Alexandria, VA 22313-1450

Hand delivered responses should be brought to:
Customer Service Window
Randolph Building
401 Dulany Street
Alexandria, VA 22314

Any inquiry concerning this communication or earlier communications from the examiner should be directed to DANIEL MITCHELL whose telephone number is (571)270-5307. The examiner can normally be reached on Monday - Friday 8:00 am - 5:00 pm EST.

If attempts to reach the examiner by telephone are unsuccessful, the examiner's supervisor, Chirag G. Shah can be reached on 571-272-3144. The fax phone number for the organization where this application or proceeding is assigned is 571-273-8300.

Information regarding the status of an application may be obtained from the Patent Application Information Retrieval (PAIR) system. Status information for published applications may be obtained from either Private PAIR or Public PAIR. Status information for unpublished applications is available through Private PAIR only. For more information about the PAIR system, see http://pair-direct.uspto.gov. Should you have questions on access to the Private PAIR system, contact the Electronic Business Center (EBC) at 866-217-9197 (toll-free). If you would like assistance from a USPTO Customer Service Representative or access to the automated information system, call 800-786-9199 (IN USA OR CANADA) or 571-272-1000.

/D. M./
Examiner, Art Unit 2477

/Chirag G Shah/
Supervisory Patent Examiner, Art Unit 2477